CA-IDMS®/DDS

Design and Operations

15.0

Computer Associates™

This documentation and related computer software program (hereinafter referred to as the “Documentation”) is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. (“CA”) at any time.

THIS DOCUMENTATION MAY NOT BE COPIED, TRANSFERRED, REPRODUCED, DISCLOSED, OR DUPLICATED, IN WHOLE OR IN PART, WITHOUT THE PRIOR WRITTEN CONSENT OF CA. THIS DOCUMENTATION IS PROPRIETARY INFORMATION OF CA AND PROTECTED BY THE COPYRIGHT LAWS OF THE UNITED STATES AND INTERNATIONAL TREATIES.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION “AS IS” WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

THE USE OF ANY PRODUCT REFERENCED IN THIS DOCUMENTATION AND THIS DOCUMENTATION IS GOVERNED BY THE END USER'S APPLICABLE LICENSE AGREEMENT.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with “Restricted Rights” as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227.7013(c)(1)(ii) or applicable successor provisions.

First Edition, December 2000

© 2000 Computer Associates International, Inc.
One Computer Associates Plaza, Islandia, NY 11749
All rights reserved.

All trademarks, trade names, service marks, or logos referenced herein belong to their respective companies.

Contents

What this manual is about

This manual applies to CA-IDMS Release 15.0 Distributed Database System (DDS). It provides information necessary to establish and maintain a database distributed among two or more sites.

Who should use this manual

This manual is intended for system and database administrators in charge of setting up and maintaining distributed databases.

How this manual is organized

This manual contains four chapters.

- **Chapter 1** presents an overview of distributed data processing, networks, and the DDS facilities for generating and operating a DDS network.
- **Chapter 2** gives a detailed discussion of design and implementation considerations.
- **Chapter 3** presents the system components in a CA-IDMS/DC or CA-IDMS/UCF system generation that define the characteristics of the DDS network.
- **Chapter 4** discusses:
  - System startup
  - Facilities for managing the network
  - Processing of application requests
  - Backup and recovery considerations
- **Appendix A** describes DDS VTAM considerations and gives sample statements for defining a DDS network using VTAM.

How product names are referenced

This manual uses the term CA-IDMS to refer to any one of the following CA-IDMS components:

- CA-IDMS/DB — The database management system
- CA-IDMS/DC — The data communications system and proprietary teleprocessing monitor
- CA-IDMS/UCF — The universal communications facility for accessing CA-IDMS database and data communications services through another teleprocessing monitor, such as CICS
- CA-IDMS/DDS — The distributed database system

This manual uses the terms DB, DC, UCF, DC/UCF, and DDS to identify the specific CA-IDMS component only when it is important to your understanding of the product.

Related documentation

For further information related to this manual, refer to:

- *CA-IDMS Database Administration*
- CA-IDMS installation manual for your operating system
- *CA-IDMS System Generation*
- *CA-IDMS System Operations*
- *CA-IDMS System Tasks and Operator Commands*
- *CA-IDMS Utilities*

Chapter 1. Distributed Data Processing, Networks, and DDS

1.1 Overview

As information requirements become increasingly complex, organizations need to find new ways to distribute processing power, application programs, and data. Distributed data processing allows distribution of application programs and data among interconnected sites to satisfy the information needs of the organization. Depending on its requirements, an organization may choose to centralize or decentralize its data processing systems.

Centralized data processing: In a centralized system, one machine controls all file access and updates.

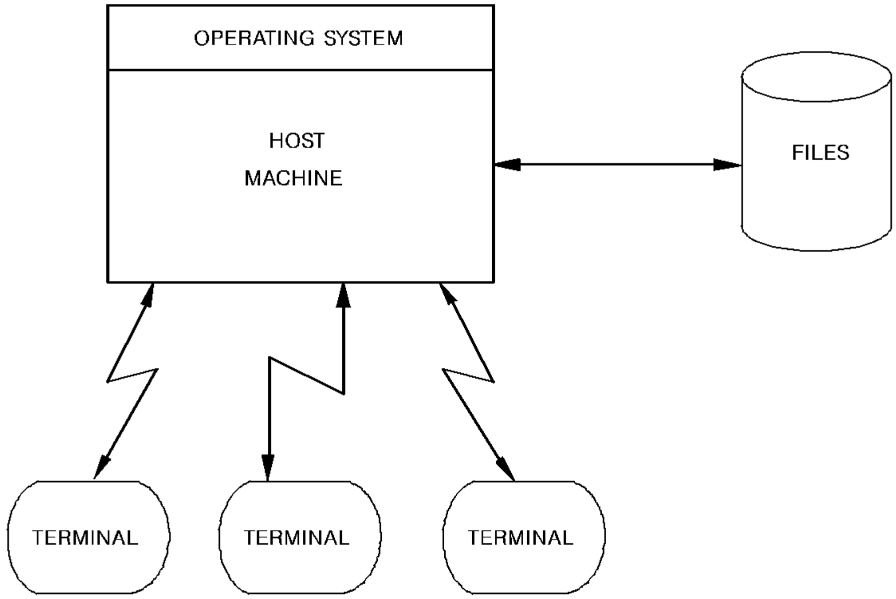
A centralized system responds to the needs of the organization by permitting a high level of control over application programs and data.

A centralized data processing system is useful when:

- All data is shared across application programs.
- Many end users need access to the same data and also require the most current data available.
- Security has been established as the responsibility of the central site.
- Large volumes of data are involved. In this case, centralized control can help handle the volume of data efficiently.

In a centralized data processing environment, more attention typically is given to direct access storage devices (i.e., their cost and reliability) than to data transmission.

The following figure illustrates the way in which one machine controls all file access and updates in a centralized system.



Decentralized data processing: In a decentralized system, multiple machines control file access and updates to serve the varied needs of end users.

End users tend to have more control over application development and operations in a decentralized environment. A decentralized system is useful when:

- Data is primarily accessed at a single location. In this case, it is most efficient to store the data where it is used.
- The update rate is high. In this case, distribution of the data across different sites would permit more efficient update operations.
- Security has been established as a local responsibility.
- Data is used in highly specialized ways by end users.

In a decentralized data processing environment, more attention typically is given to data transmission costs and reliability than to direct access storage devices.

Two kinds of networks: Distributed data processing networks make it possible to combine the benefits of both centralized and decentralized systems. These networks can be viewed in two ways:

- **Physical network** — The combination of interconnected equipment (hardware) and programs (telecommunications access methods) used to transmit data between physical locations.
- **Logical network** — The combination of databases and application programs that handle an organization's information and processing requirements.

DDS allows distributed processing: The CA-IDMS Distributed Database System (DDS) is designed to respond to the need for distributed processing. The remainder of this chapter presents an overview of DDS concepts and network configuration.

1.2 DDS concepts

This section describes DDS concepts.

Each system a node: DDS is a system that establishes and maintains communication between nodes in a distributed data processing network. Each node in the network is a DC system, or a Universal Communications Facility (UCF) system. Each node can be generated to provide the services required by a particular group of applications or a wide range of different applications.

Note: The term *node*, when used in this manual, refers to a DDS node (a DC system, or a UCF system), and should not be confused with a VTAM node or an SNA node.

Using DDS, an organization can realize the benefits of both centralized and decentralized data processing. Nodes can be located at the same site or at remote sites, and each node can control one or more CA-IDMS databases. Application programs can access any database in the network, and the network is designed so the location of all databases is transparent to application programs.

Host and target nodes: Under DDS, a single node accepts application requests for database services and either processes those requests directly by accessing a database under its controls or passes the requests to the node that controls the appropriate database. Once the requesting node (the *host* node) routes a request to another node (the *target* node), the target node performs the requested service and returns the results to the host. The host then returns the results to the application. The process of locating the appropriate database and routing the requests to the controlling node is transparent to the requesting application.

Distribute or centralize data: The data available to applications executing within the DDS network can be distributed among databases controlled by several different nodes; alternatively, all data can be stored centrally in a single database, where the controlling node processes all application requests originating at other nodes. Additionally, using UCF, the network can be designed to permit online transactions originating at one node to be executed at a remote node.

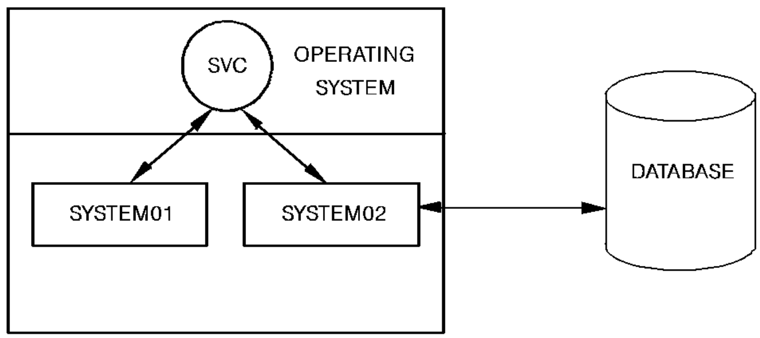
DDS is flexible: DDS permits flexibility in designing a distributed database network. Any number of DC/UCF systems can participate in the network; each node can be located at a remote site, or several nodes can be located at a single site. Depending on the location of the individual nodes, communication between nodes takes place by way of the CA-IDMS SVC (Supervisor Call) Routine or a telecommunications line.

1.3 DDS network configuration

A DDS network consists of multiple nodes connected by the CA-IDMS SVC or a telecommunications line, as follows:

- **CA-IDMS SVC** — The CA-IDMS SVC connects two or more nodes residing in a single machine, as shown in the following figure.

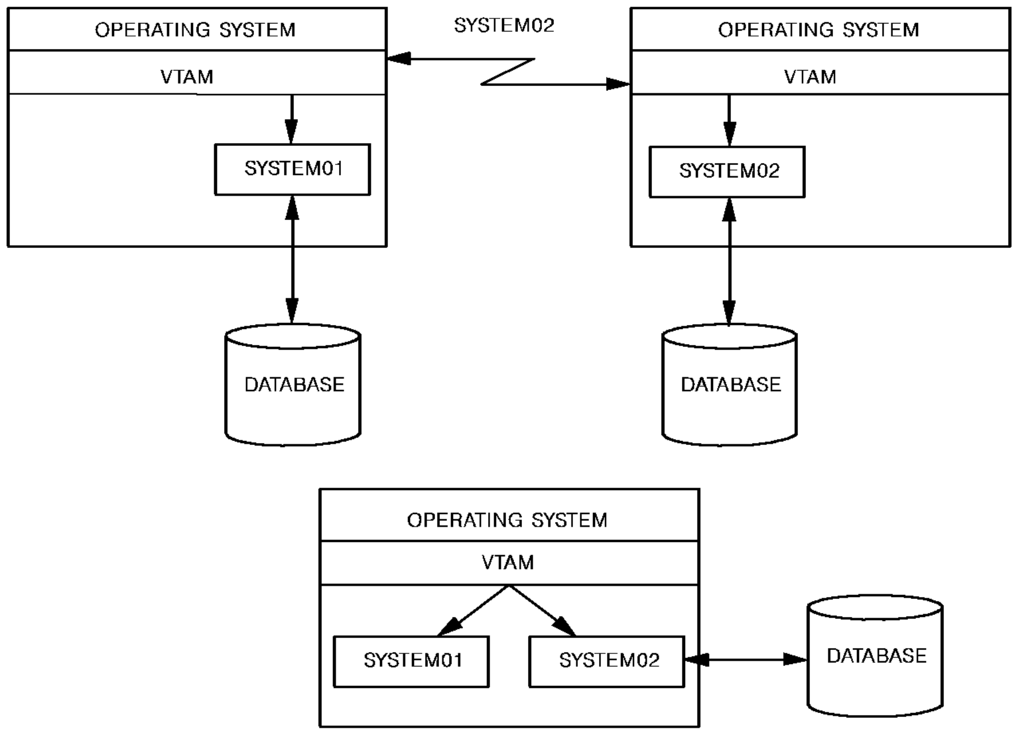
  Data and application requests are transferred between the two nodes through the SVC.



- **Telecommunications line** — A telecommunications line connects two nodes residing in different machines at remote sites or two nodes in the same machine. Data and application requests are transferred between the two nodes across the line by the telecommunications access method. The telecommunications lines between nodes can be any supported by the Common Communications Interface (CCI) or VTAM.

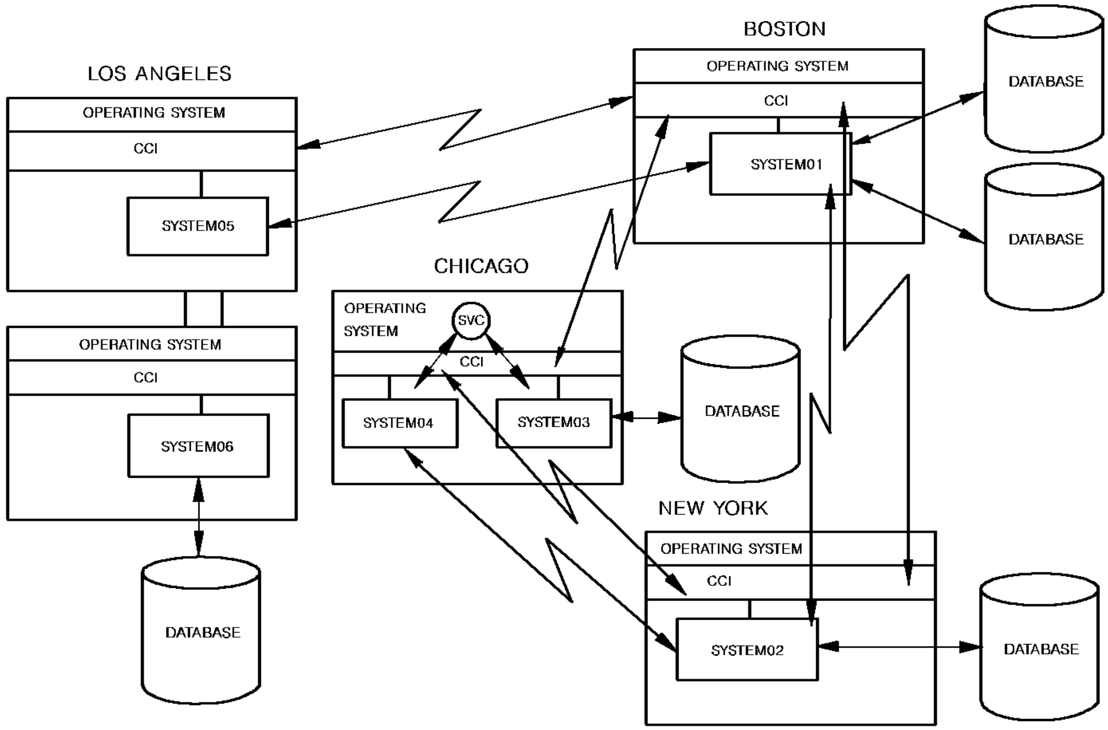
  ►► For a list of supported communications access methods, refer to *Unicenter TNG Framework for OS/390 Getting Started or Administrator Guide*.

  VTAM — VTAM controls communications between machines. When VTAM is used to connect two DDS nodes, communication is controlled by VTAM. Use of VTAM in a DDS environment is illustrated below.



Direct and indirect connections: In a DDS network, a node can be connected directly (by way of SVC or a teleprocessing line) to any number of other nodes. If two nodes are connected directly, communication between them can take place by way of this direct connection. If two nodes are not directly connected, communication between them takes place by way of one or more nodes that are directly connected to the host and target nodes. The following figure illustrates a sample DDS network.

The nodes in New York and Boston can communicate by way of a direct connection. However, node SYSTEM06 in Los Angeles would communicate with node SYSTEM01 in Boston by way of SYSTEM05.



Configuring the network: The DDS network configuration is defined through DC/UCF system generation procedures. In addition to standard system generation entries, the system generation of each system that participates in the network includes entries to:

- Identify the node
- Define the node's direct connections to other nodes
- Name the databases controlled by the node

The user can obtain reports on the generation of nodes through the Data Dictionary Reporter (DDR).

►► For more information about DDS system generation, refer to Chapter 3, “DDS System Generation” on page 3-1.

Changing configuration: Facilities are available that allow the user to alter the network configuration as necessary.

►► Refer to Chapter 4, “DDS System Operations” on page 4-1, for more information about managing the network.

Path selection: Within a DDS network, DDS controls the selection of the path along which application requests are transferred between nodes. DDS path selection and global and local databases are discussed in Chapter 2, “Network Design Considerations” on page 2-1.

The remainder of this manual provides information essential to the design and implementation of a DDS network. The topics are:

- Network Design Considerations
- DDS System Generation
- DDS System Operations

Chapter 2. Network Design Considerations

2.1 Overview

The initial task of the DDS administrator is to analyze the organization's information requirements. By using the results of this analysis, the administrator can begin to define the characteristics of the network. The structure of the DDS network should reflect the way information is handled by the organization and allow the organization to access the information as efficiently as possible.

This chapter offers an approach to designing a DDS network from an organization's informational requirements. The stages of the design process are:

- **Conceptual specification** — Analyze and define the organization's information requirements.
- **Design specification** — Adapt the conceptual specification to DDS requirements.
- **Implementation specification** — Develop the design into a DDS network by defining nodes, paths, and databases.

2.2 Conceptual specification

This section describes how to create a conceptual specification for a DDS network.

Analyze the organization: The initial conceptual specification for a DDS network should be removed from DDS design and implementation considerations. The conceptual specification should fall naturally from the way the organization handles information.

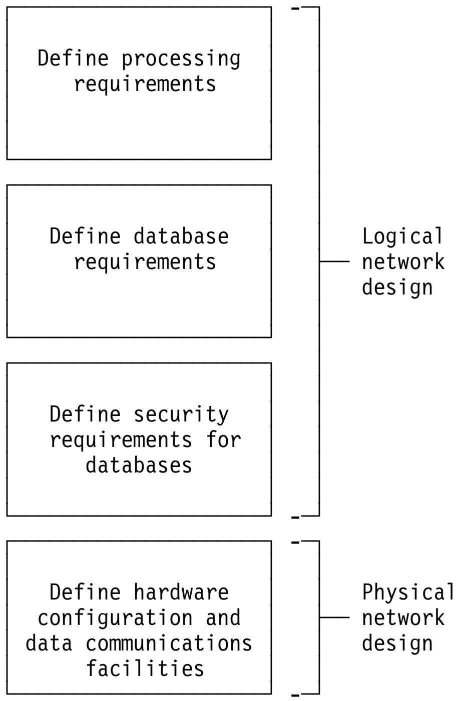
Since the primary requirement for a sound conceptual specification is knowing the organization's information requirements, an organization may decide to coordinate this phase of network design with business analysts, who may not know a great deal about computers, but do know the organization.

Take the following steps to create a conceptual specification:

1. Identify and examine applications; also, develop a catalog of software components.
2. Identify and examine databases, both existing and potential.
3. Define security requirements for databases.
4. Analyze and evaluate hardware configurations and data communications facilities.

As the following figure illustrates, steps 1, 2, and 3 define the logical network, and step 4 defines the physical network. The conceptual specification of a DDS network is based on the analysis of both the logical and physical networks.

These four steps are discussed separately below.



2.2.1 Identifying and examining applications

This section describes how to identify and examine applications that will run in a DDS network.

Define processing requirements: Identifying and examining the application programs and other software components that access the organization's databases allows the DDS administrator to determine the processing requirements of the organization. The execution of these steps may vary from site to site.

- Identify users of the organization's data, both people and programs.
- Analyze the frequency of data access and identify the times that certain data is likely to be accessed.
- Determine the volume of data being accessed.
- Identify the individuals (or sites) responsible for maintaining applications.

Defining the organization's processing requirements helps the DDS administrator determine where the application programs are most needed.

Because processing requirements constantly change and grow, these steps are likely to be repeated. By keeping abreast of changing processing requirements, the DDS administrator can accurately assess the efficiency of the network and decide when modifications are needed.

2.2.2 Identifying and examining databases

This section describes how to identify and examine databases that will be accessed in a DDS network.

Define database requirements: Identifying and examining the organization's databases allows the DDS administrator to determine what data needs to be accessed and where the data is currently maintained. Security requirements for the organization's databases should also be examined because the databases participate in a network and can be accessed from any node. The DDS administrator can secure these databases using standard CA-IDMS security features.

►► Refer to *CA-IDMS Security Administration* for details about database security.

Examining the organization's databases provides the DDS administrator with an understanding of the data requirements of the organization. This information, coupled with a knowledge of the processing requirements (determined by examining applications), leads to a logical network design specification, as discussed later in this chapter.

2.2.3 Analyzing hardware configurations

This section describes how to analyze your hardware configurations.

Define existing physical network: Existing hardware configurations and data communications facilities must be considered when creating the initial conceptual specification. The hardware configuration and data communications facilities determine the physical network design. The DDS administrator must be familiar with the existing configuration and be able to reconfigure the system in order to accomplish the level of distributed processing required by the organization.

2.3 Design specification

Once the characteristics of the application programs (processing requirements) and databases (data requirements) have been identified, the DDS administrator can begin to design the network. DDS network design involves using the facilities available with DDS to construct a network that most efficiently organizes data and application programs and places system resources close to the users of those resources.

When designing a DDS network, the system administrator should consider:

- Operating systems and telecommunications access methods
- Methods of connection
- Path selection
- Host and target databases

2.3.1 Operating systems and telecommunications access methods

A DDS network can operate on any environment that CA-IDMS and either CCI or VTAM support.

▸▸ For details about available telecommunications methods in the DDS environment, see *Unicenter TNG Framework for OS/390 Getting Started or Administrator Guide*.

2.3.2 Methods of connection

Each node in a DDS network is connected to at least one other node. Three types of connections can exist:

- **CA-IDMS SVC (Supervisor Call) Routine** — The CA-IDMS SVC connects two nodes that reside in the same machine.
- **Virtual Telecommunications Access Method (VTAM)** — VTAM connects two DDS nodes on different machines (or the same machine) and controls communication between the nodes.
- **CCI** — CCI connects two nodes on different machines or on one machine.

Typically, the types of connections chosen depend on the hardware and data communication facilities available at the site.

2.3.3 Path selection

In a DDS network configuration, several paths can exist between two nodes. The nodes can be connected using CCI or VTAM.

Using CCI: When a host node passes a database request to a target node, CCI directs the communication between the two nodes by selecting the optimum path along which to pass the request. The path selected by CCI may be a direct path between the two nodes, or, if the nodes are not directly connected, may be by way of one or more intermediate nodes.

The path selected is used to pass all application requests to the target node. If the path becomes unavailable (for example, if a telecommunications line fails or if a node in the path signs off from the network) during application execution, CCI automatically selects another available path, usually without interruption in processing.

Using VTAM: When using VTAM, the host node and the target node will have previously connected to each other, and the request is sent on that connection.

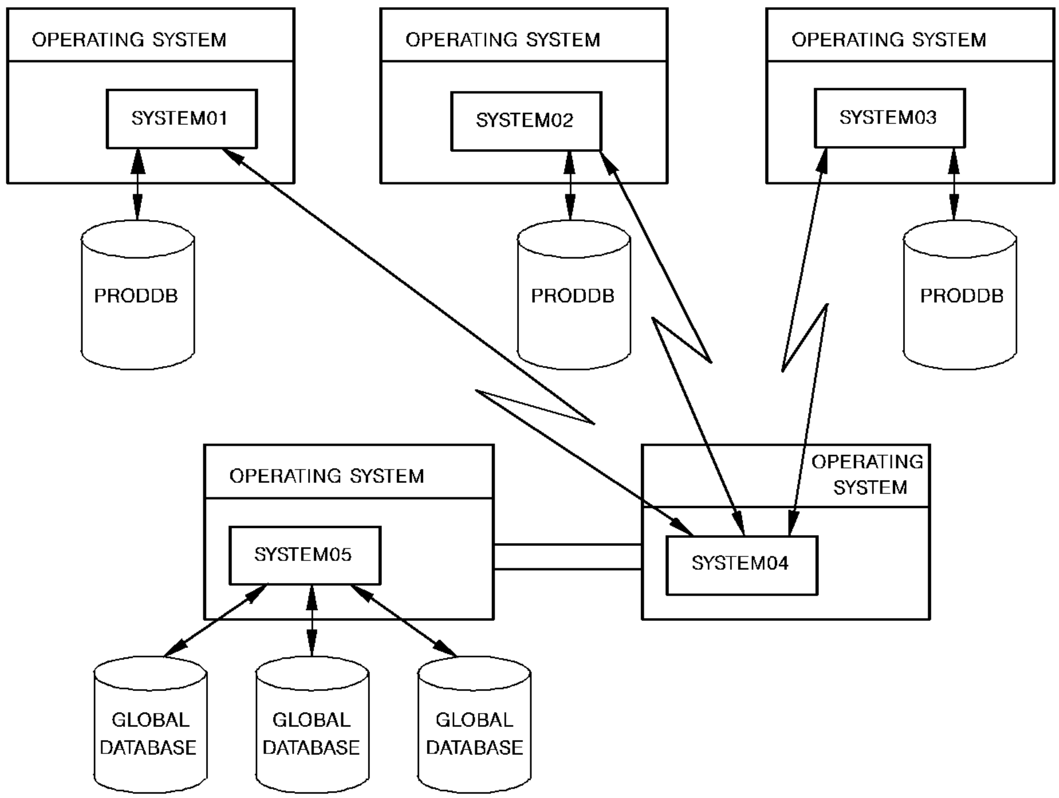
2.3.4 Host and target databases

A node can control one or more databases. Each node maintains a table (the resource name table) that lists the database(s) it has access to, their locations, and how to reach them. This table is maintained by DDS at the node, allowing the location of any database to be transparent to applications.

Database names can be duplicated within the network. Each node can control a database that replicates local databases controlled by other nodes in the network. Each node's table would simply define the location of the database as local, and applications asking for that database would access the local one.

The choice between local and global databases should be based on how the databases are accessed. For example, if each site in a DDS network maintains its own production database, those databases could each have the same name and be defined as local databases. However, if the organization maintains a central database required by multiple sites, it would make sense to define that database as global.

The following figure illustrates a DDS network configuration that maintains local production databases at distributed sites while also maintaining centrally-located global databases.


OPERATING SYSTEM
SYSTEM01
PRODDB
OPERATING SYSTEM
SYSTEM02
PRODDB
OPERATING SYSTEM
SYSTEM03
PRODDB
OPERATING SYSTEM
SYSTEM05
GLOBAL DATABASE
GLOBAL DATABASE
GLOBAL DATABASE
OPERATING SYSTEM
SYSTEM04

2.4 Implementation specification

At this point, the DDS administrator should have a clear idea about the organization's data requirements, processing requirements, and facilities available through DDS to distribute the organization's databases and applications.

The DDS administrator can begin to implement the network design by creating a network diagram that can be used as a basis for coding the necessary DC/UCF system generation statements. This process involves defining node and connection characteristics:

- **Node Characteristics** — Name and identification of the node, operating environment, and associated databases
- **Connection Characteristics** — Type of connection and path identification

2.4.1 Connecting nodes

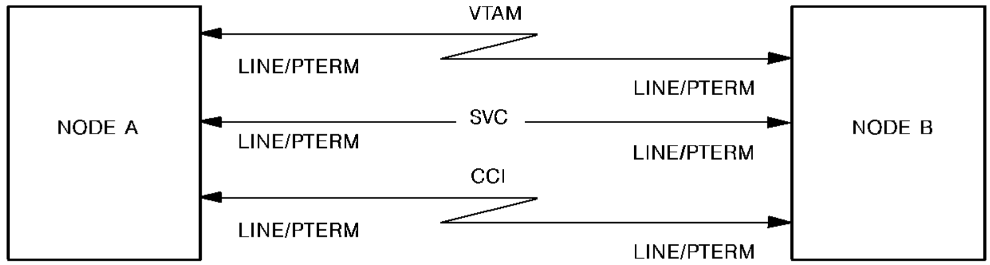
This section describes how nodes are connected in a DDS network.

Ports: Under DDS, a connection between two nodes is defined in terms of a port. A port is an access point through which the node passes request and response packets to another node. A port is defined at system generation time by the resource name table (IDMSCSTB).

►► For details about ports see Chapter 3, “DDS System Generation” on page 3-1.

The resource name table tells the node the names of other nodes and the path used to reach each other node. For example, suppose node A is connected to node C by way of node B. To communicate with node C, node A sends a message *to* B *for* C. B receives the message, and forwards it to C. All A needs to know is that messages to C should be sent to B. That information is maintained in the resource name table.

Two types of connection: A DDS connection is represented in the DDS network diagram by one of two different types of lines, depending on the type of connection between two nodes. A straight line represents a CA-IDMS SVC connection, and a communication link represents either a VTAM or a CCI connection. The LINE/PTERM pair that defines the node's port is placed beneath the line and near the node that uses the port. The following figure illustrates the types of DDS connections and the information associated with the connections.



Coding node definitions: The resulting DDS network diagram can be used as a basis for coding the DC/UCF system generation statements needed to define the node.

►► For details about coding these statements, see *CA-IDMS System Generation*

►► A discussion of the DDS components included in a DC/UCF system generation is presented in Chapter 3, “DDS System Generation” on page 3-1.

Chapter 3. DDS System Generation

3.1 Overview

This chapter describes the DDS components included in a DC/UCF system generation to define a node. For system generation syntax and rules, see *CA-IDMS System Generation*.

Each node specified: Each node in a DDS network is generated by including specific DDS system generation statements and clauses with the standard DC/UCF system generation statements. These statements and clauses allow the DDS administrator to specify the characteristics of the node and its relationship with other nodes. During DC/UCF system generation, the individual node, its direct connections to other nodes, system control information, and database information are defined:

- **Node name** — The name by which the node is known to other nodes in the network.
- **DDS ports** — The LINE and PTERM pair that defines the access point through which the node passes request and response packets to other nodes.
- **SYSCTL Overrides (OS/390)** — System control information used to direct external request units to the appropriate node and/or database.

The DC/UCF system generation statements include:

```
ADD LINE CCILINE
    TYPE IS CCI.
ADD PTERM CCIPT01
    IN LINE CCILINE
    TYPE IS BULK.
ADD LTERM CCILT01
    ENABLED
    PTERM IS CCIPT01.
```

DDS reports: System generation reports that produce DDS-specific information are available. These reports allow the DDS administrator to examine and evaluate the components of the DDS network.

A discussion of DDS system characteristics and reporting capabilities follows.

3.2 Node name

Each node must have a name by which it is known to all other nodes in the network. A node can have only one name associated with it, and that name must be unique throughout the network. The node name is not necessarily the same name as is used to sign on to the system. For example, to invoke a DC system from a VTAM terminal, the following command might be used:

```
IDMSDC
```

If the system invoked by the above command had a version number of 8, and a node name was not specified, the node would be given a default name of SYST0008. Thus all other nodes would know it as SYST0008. It is important not to confuse the task code or signon command with the node name.

Establish naming conventions: Naming conventions should be established for nodes to ensure uniqueness and clarity. One approach (as used in this manual) would be to name all nodes by using the system number specified in the system generation SYSTEM statement. For example, SYSTEM01 and SYSTEM02 would be the node names for system numbers 01 and 02.

Note: The node names default to SYST0*nnn*. See the SYSTEM statement description in *CA-IDMS System Generation*.

3.3 DDS ports

A DDS port is defined during system generation with a LINE and PTERM pair. A port represents an access point through which the node passes request and response packets to another node. Each node in a network can be connected to any number of other nodes.

DC/UCF system generation statements include clauses specific to DDS that allow the DDS administrator to specify characteristics about the port. Through the definition of these ports the DDS administrator defines some of the characteristics of the network and controls data transmission through the network.

The characteristics of DDS LINEs and PTERMs are discussed separately in the following paragraphs.

3.3.1 LINEs

CCI LINE: CCI LINEs are used by the node to establish communication between nodes. The parameter of the CCI LINE statement is TYPE IS CCI, which identifies the line as a CCI line that will be used to connect two DC/UCF systems (nodes).

For more information about CCI LINEs, see *CA-IDMS System Generation*.

DDS LINE: DDS LINEs are used by the node to establish communication between nodes. DDS LINEs should not be confused with other DC/UCF LINE types, or with telecommunications lines. The two parameters of the DDS LINE statement are:

- TYPE IS DDS indicates that the LINE is to be used to establish a connection between two nodes.
- The SOURCE parameter (VTAM only) specifies the application program identification used to define the DDS node to VTAM. VTAM uses this information to establish connection between two DDS nodes.

For more information about VTAM in the DDS environment, see Appendix A, “DDS VTAM Considerations” on page A-1

3.3.2 PTERMs

CCI: CCI PTERMs are used in conjunction with CCI LINEs to establish communication between two nodes. The CCI PTERM parameter is TYPE IS BULK, which specifies that the data transfer between two DC/UCF systems will be bulk.

DDS: DDS PTERMs are used in conjunction with DDS LINEs to establish communication between two nodes. The DDS PTERM defines the node's line driver and supplies parameters to be used by the line driver at run time. Typically, multiple PTERMs are defined for each LINE. DDS PTERMs, unlike DC/UCF physical terminals, do not need to be associated with LTERMs.

A DDS PTERM is specified at system generation with TYPE BULK or TYPE VTAM.

TYPE VTAM: You specify VTAM as the PTERM TYPE if the connection being established is between two nodes that reside on different machines connected by a VTAM communications link. The following additional PTERM parameters can be specified when using TYPE VTAM:

- **BLOCKSIZE** — Specifies the size of the packet used to pass data between nodes.
- **TARGET=***VTAM application-id* — Specifies the VTAM LUNAME of the target DDS node. VTAM *application-id* corresponds to a VTAM minor node name and must be unique throughout the network. This is a required parameter.
- **WEIGHT FACTOR** — Specifies the priority to be assigned to the port defined by the PTERM and its associated line. When selecting a port through which to pass request and response packets, the node bases its selection on the PTERM weight factor for each port that connects the node to a target node.

3.4 Resource name table

The ability to access multiple databases is inherent in DDS. If multiple databases are to be controlled by the node, an entry for each database to be accessed must exist in the node's resource name table.

For convenience, generic names using wild cards are supported. Thus, PROD* could be used to specify all databases with names beginning with PROD.

3.5 System generation reports

DC/UCF system generation reports are available that produce DDS-specific information. These reports fall into two categories:

- **Source reports** — System generation information resulting from system generation phase 1. These reports describe the generated forms of DDS entities.
- **Sysgen reports** — System generation information resulting from system generation phase 2. These reports describe the executable forms of DDS entities.

These reports are available as CREPORTS through Data Dictionary Reporter (DDR).

►► For specific information about running CREPORTS, see *CA-IDMS Reports*.

►► For samples of all DC/UCF system generation reports, see *CA-IDMS Reports*.

The following table lists the standard CREPORTS available for reporting on DDS definitions in the data dictionary.

Report Number	**Report title**	**Data dictionary module name**	**Category**
01	Network description by line	CREPORT 001	SYSGEN
02	Network description by physical terminal	CREPORT 002	
14	Network description by line	CREPORT 014	SOURCE
15	Network description by physical terminal	CREPORT 015	
16	Physical terminals within line	CREPORT 016	
43	DDS report	CREPORT 043	SYSGEN *and* SOURCE

Chapter 4. DDS System Operations

4.1 Network membership affects operations

The operation of a DDS node is similar to that of any DC/UCF system. The node is started up in the same way and, for the most part, executes like a standard DC/UCF system. The differences that do exist are the result of the communication that takes place between the nodes. Because these systems participate in a network and can access databases other than their own, there are new considerations concerning how the network is managed, how application requests are processed, and how backup and recovery are performed.

This chapter presents operational considerations for using DDS. The following topics are discussed:

- DDS system startup
- Facilities for managing the network
- Processing application requests
- Backup and recovery

This information applies to the DDS environment only; it is intended to supplement the DC/UCF operational procedures, as discussed in the following publications:

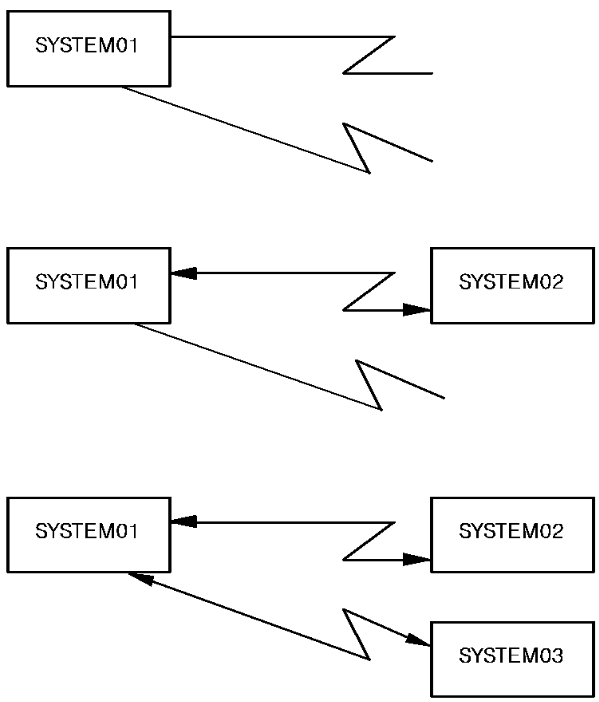
- *CA-IDMS Database Administration*
- *CA-IDMS System Generation*
- *CA-IDMS System Operations*

4.2 DDS system startup

The DDS startup procedure is identical to the standard DC/UCF startup procedure. When a node is started up, it transmits messages through its ports to the other nodes to which it is connected. If the other node is active, a connection is established and information is exchanged between nodes. This information is stored internally as tables maintained by DDS.

Each node knows the status of neighbors: As nodes sign on or off, these tables are updated to reflect the changes in the network. At any point while the network is operating, every node in the network knows the status of all the nodes to which it maintains direct connections.

The following figure illustrates the sequence of events that takes place when a DDS network is started up.



4.2.1 Path maintenance

In a DDS network, multiple paths can exist between nodes; if one path becomes unavailable, DDS selects another path to maintain communication between nodes.

VTAM users, however, can alter the path selection by changing the weights associated with a DDS port. See *CA-IDMS System Generation* for a discussion of the WEIGHT FACTOR DDS PTERM parameter.

4.3 Facilities for managing the network

Once a DDS network is in operation, it is necessary to monitor the activity of the network and make alterations as needed. Facilities for managing the DDS network are available in these forms:

- **CCI** — Allows addition and deletion of telecommunication paths.
- **DCMT functions** — Allow the user to monitor the network and make immediate changes as needed.
- **Statistics reports** — Offer a more complete, in-depth analysis of the network.

4.3.1 DCMT functions

DC/UCF master terminal functions provide the means to monitor and control the various aspects of the DDS system at run time:

DISPLAY DDS	Displays general DDS information including information about nodes, lines, and physical terminals (PTERMs).
DISPLAY LINE	Displays information about a line.
DISPLAY PTERM	Displays information about a PTERM.
VARY LINE	Enables the user to vary the line online or offline.
VARY PTERM	Enables the user to vary the terminal online or offline.

►► For more information about the DCMT functions in the above table, see the *CA-IDMS System Tasks and Operator Commands*.

VARY commands should be secured: The DCMT VARY LINE/PTERM commands should be used only to vary the line and/or physical terminal online or offline. Since use of this command could affect network operations, these tasks should be adequately secured. DC/UCF permits the assignment of a discrete security class to each of the DCMT commands, thereby allowing the DDS administrator to limit the use of the VARY command to authorized persons only. These DCMT functions apply directly to DDS system operations.

►► For more information about discrete security for the DCMT task, see *CA-IDMS Security Administration*.

►► For more information about DCMT functions applicable to the DC/UCF environment, see the *CA-IDMS System Tasks and Operator Commands*.

4.3.2 Statistics reports

DC/UCF statistic reports (SREPORTS) are available to summarize statistical data written to the log area (DDLDCLOG) during system execution and subsequently offloaded (by means of the ARCHIVE LOG utility) to an archive file. These reports allow the DDS administrator to examine network activity. Using these reports, the administrator can make informed decisions about the efficiency of the network. The following table lists the DC/UCF statistics reports available.

▸▸ For further discussion of system statistics, see *CA-IDMS System Tasks and Operator Commands* or *CA-IDMS Reports*.

Number	**Description**
000	Performs housekeeping functions for the other reports and produces no output. This report must be copied into all statistical report runs.
001	Produces system and external run-unit (task) histograms.
002	Produces line histograms applicable for DDS.
003	Produces general system-wide statistics.
008	Produces external run-unit (task) statistics by accounting data.
009	Produces external run-unit (task) statistics by program name.
012-016	Produce system-wide external run-unit (task), program, line, and physical terminal statistics. Reports 012 through 016 must always be requested together.
099	Produces an output file of all statistics records contained in the input archive log file, but performs no formatting; thus, the records in the output file are identical in format to the records in the archive log file. Since it contains statistics records only, the output file is smaller than the archive log file and provides a convenient means of saving statistics records for input to future statistical report runs.

4.4 Processing application requests

Processing application requests for database services in the DDS environment involves:

1. Identifying the target database and node
2. Routing database requests to the target node
3. Servicing database requests

Identifying the target database and node: When a request unit signs on to its host node, it names a target database and/or node. This action signals to the host node that all requests for database services are to be routed to the named node or database. If the unit names only a target node and no database, the target node selects the default database (under its control) as the one to be accessed to service requests issued by the request unit. If the request unit names the host node or a database controlled by the host node, all requests for database services are handled by the host node.

Routing database requests to the target node: After the target node has been identified, DDS selects the optimum path along which to route database requests from the host node to the target node. The selected path is used to transfer every database requests issued by the request unit. If the path becomes unavailable at any point during request unit processing, DDS automatically selects another path if one is available.

Servicing database requests: When the request unit issues a database access request, the host node signals DDS to transfer the request and any accompanying data to the target node. DDS routes the request and data along the selected path to the target node. The target node performs the requested database operation and returns a response to the host node, which in turn passes the results of the request to the request unit.

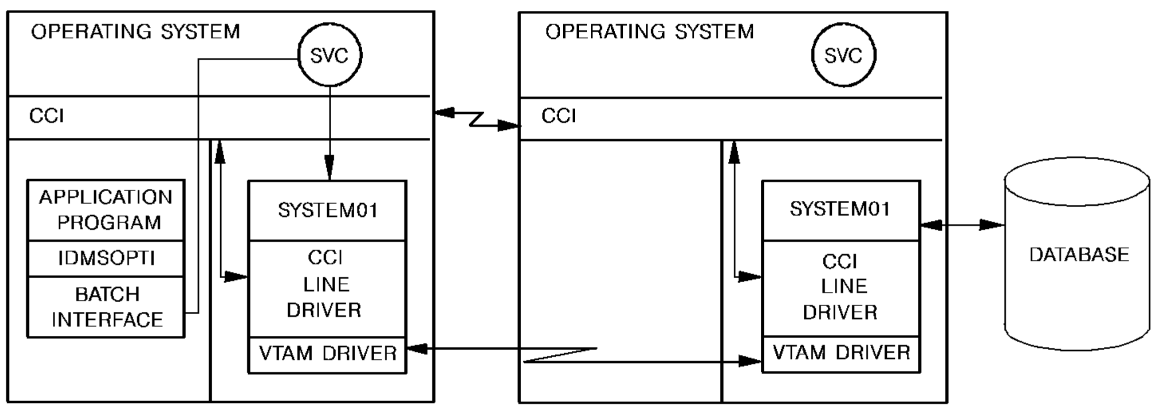
Target nodes do the work: Each database request issued by the request unit is serviced in the same manner: the host node passes a request to the target node, which services the request and returns a response to the host node. If, at any point during run-unit execution, the application program abends, the target node performs any necessary recovery of the database being accessed. When the request unit terminates normally, the application program can begin a new request unit.

When an application running under SYSTEM01 requests database services from SYSTEM02, the following sequence of events takes place:

1. The request is passed from the CA-IDMS batch interface, called by the application program, to the host node (SYSTEM01).
2. The host node builds a database request packet and posts the DDS line driver to transmit the packet to the target node (SYSTEM02).
3. The DDS line driver transmits the packet to its counterpart on the target node.

4. The DDS line driver on the target node receives the packet, validates the message, and passes the packet to the CV executing on the target node to process the request.
5. The target node processes the request. Database access, if required, is performed by the DBMS within the target node.
6. The target node builds a response packet with the necessary data and posts the DDS line driver to transmit the packet back to its counterpart on the host node.
7. The DDS line driver on the host node receives the packet, validates the message, and passes the packet to the task executing on the host node; the response data is then passed to the application program.

The following figure illustrates the process involved in executing a database request through DDS.

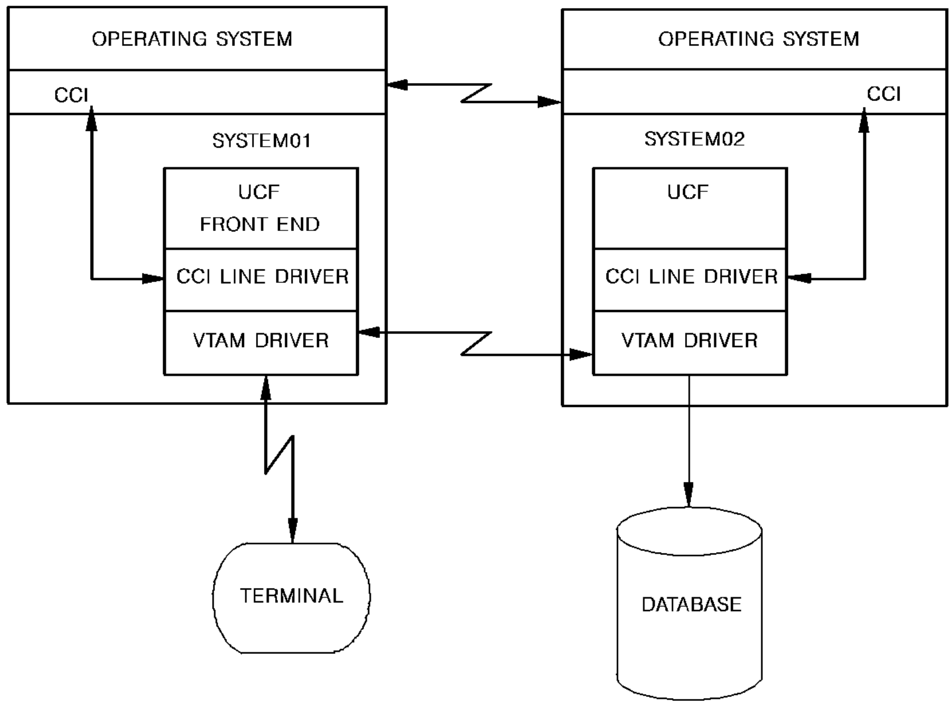


Applications executing in the DDS environment can execute multiple request units sequentially or concurrently. Each concurrently executing request unit can access a different database and/or node.

4.4.1 Remote tasks and terminal I/O

Under DDS, a target node executes tasks for a host node. When the DDS network configuration includes two or more DC/UCF systems, a user signed on to a terminal controlled by one system can request a task be executed at the target node, which is itself a DC/UCF system. In this situation, the host node performs terminal I/O operations while the target node executes the program and performs the database I/O operations or routes the request through DDS to yet another node. In environments where certain tasks requested at the host node involve considerable I/O to and from a remote database but little terminal I/O, executing the task at the node that controls the database serves to minimize the transmission of data between the host and target nodes.

When a task requested at the host node is executed at the target node, it is processed by DC/UCF tasks that execute under the target node. All database requests are issued by, and their results returned to, the tasks executing under the target node. These tasks, in turn, issue terminal I/O requests, which are passed by DDS from the target node to the host node. The host node performs the appropriate terminal I/O operations to map data from the issuing task to the terminal or to map data from the terminal to the issuing task. The following figure illustrates the remote execution of DC/UCF tasks.



4.5 Backup and recovery

Each node has available the full range of DC/UCF backup and recovery facilities (journaling, warmstart, and rollback). In the event of system or program failure, each node is responsible for the recovery of the database(s) under its control, including databases that have been accessed on behalf of other nodes.

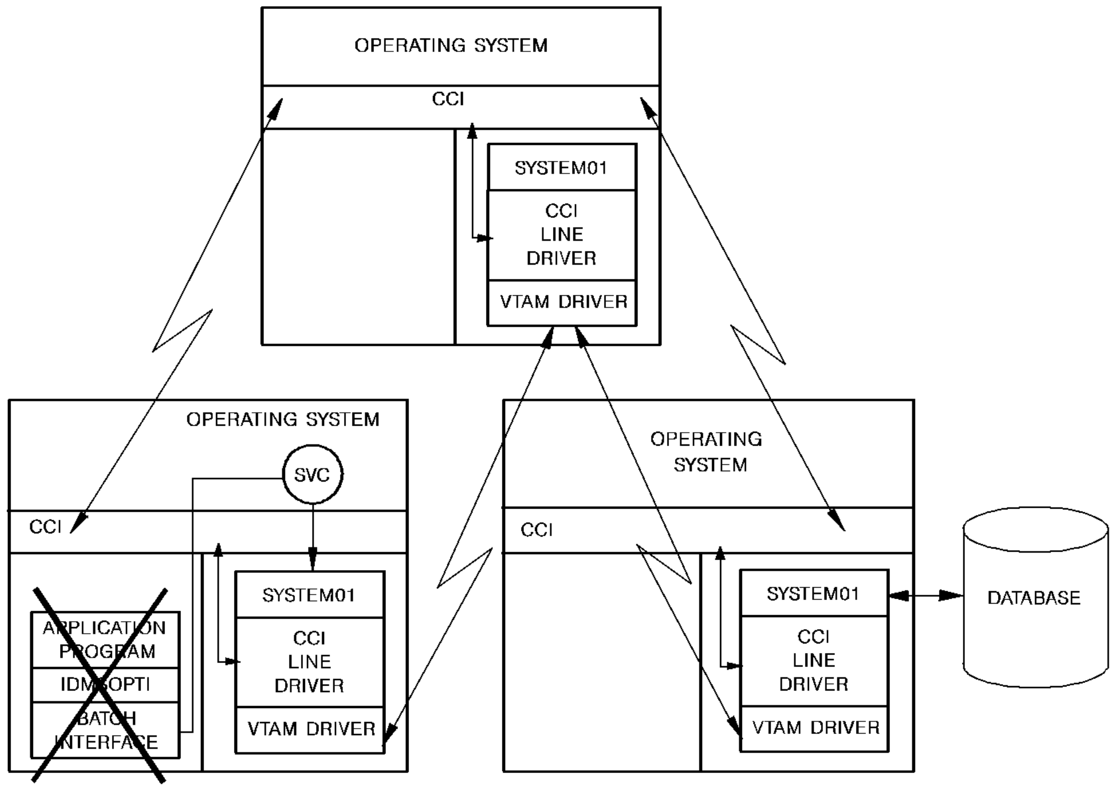
The action taken by DDS in the event of system or application program failure varies, depending on whether the failure is a system or program failure:

- **System failure** — When a node is disconnected, either because the system crashed or a connection failed, the node is responsible for recovery of all active request units under its control. Additionally, if the node provides a path for other nodes, that path is no longer available. If alternative paths are available, DDS will automatically select the best alternative path and continue processing. Restart requires only that the failed node be restarted; when restart occurs, communication between all nodes in the network is automatically reestablished.
- **Program failure** — When an application program fails, recovery takes place in exactly the same way that it does in a standard DC/UCF system. The target node handles all database recovery by using standard CA-IDMS recovery features, and the applications are restarted as appropriate.

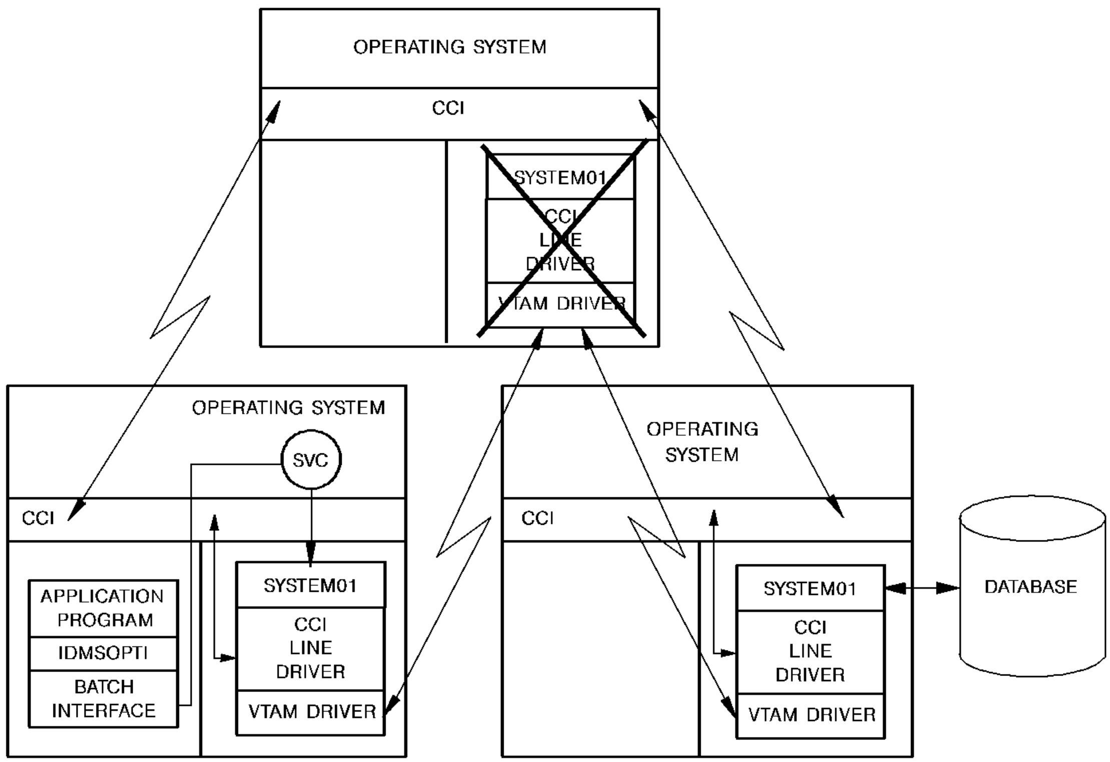
►► For a complete discussion of CA-IDMS backup and recovery facilities, see *CA-IDMS System Operations*.

The following figures illustrate how database recovery is handled in the DDS environment following an application program failure, and a system failure in the same DDS configuration.

Recovery after application program failure:



Recovery after system failure:

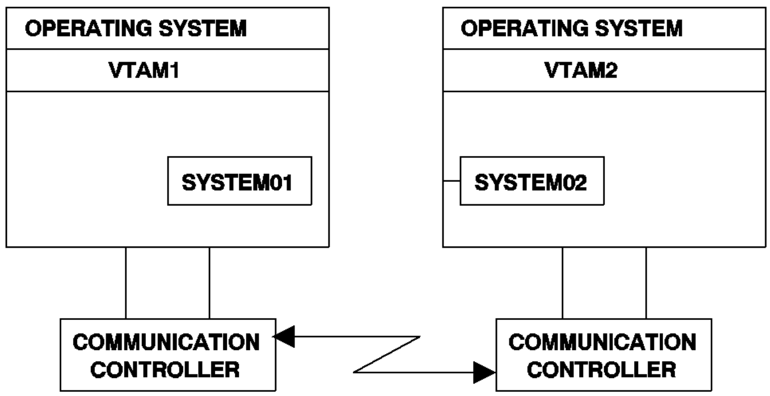


Appendix A. DDS VTAM Considerations

A.1 Overview

This appendix presents a sample DDS configuration that uses VTAM to connect two DDS nodes residing in different machines, as illustrated below. The configuration in this example is a two-domain VTAM network. VTAM with the Multisystem Networking Facility (MSNF) is running in each machine.

The following illustration shows the additional VTAM entries and CA-IDMS system generation statements needed to run in a DDS environment.



A.2 VTAM entries

The following VTAM entries are included in the VTAMLST data set for VTAM1:

```
APPL MAJOR NODE:

SYSTEM01 APPL   AUTH=(ACQ,NOPASS,NVPACE,NOTSO,SPO)
DDSNODE1 APPL   AUTH=(NOACQ,NOPASS,NVPACE,NOTSO,SPO),PARSESS=YES

CDRSC MAJOR NODE:
     DDSNODE2   CDRSC   CDRM=CDRM02
```

The following VTAM entries are included in the VTAMLST data set for VTAM2:

```
APPL MAJOR NODE:

SYSTEM02  APPL   AUTH=(ACQ,NOPASS,NVPACE,NOTSO,SPO)
DDSNODE2  APPL   AUTH=(NOACQ,NOPASS,NVPACE,NOTSO,SPO),PARSESS=YES

CDRSC MAJOR NODE;
     DDSNODE1   CDRSC   CDRM=CDRM01
```

A.3 System generation entries

The following examples show the CA-IDMS system generation statements required to define the systems in the previous illustration.

SYSTEM01 sysgen:

```
SYSTEM 01
.
.
LINE VTAM01
 ENABLED
     TYPE IS VTAMLIN
     APPL ID IS SYSTEM01...
PTERM...
.
.
LINE DDS01
 ENABLED
     TYPE IS DDS
     SOURCE=DDSNODE1.

PTERM PT01DDS1
     TYPE VTAM
     BLOCKSIZE 8192
     TARGET DDSNODE2
     WEIGHT 20.
PTERM PT02DDS1
     TYPE VTAM
     BLOCKSIZE 8192
     TARGET DDSNODEX (NOT SHOWN ABOVE)
     WEIGHT 30.

PTERM PT01BULK
     TYPE BULK.

LTERM LT01BULK
     PTERM IS PT01BULK.
```

SYSTEM02 sysgen:

```
SYSTEM02
.
.
LINE VTAM02
 ENABLED
     TYPE IS VTAMLIN
     APPL ID IS SYSTEM02....
PTERM....
.
.
.
LINE DDS02
 ENABLED
     TYPE IS DDS
     SOURCE = DDSNODE2

PTERM PT01DDS2
     TYPE VTAM
     BLOCKSIZE 8192
     TARGET DDSNODE1
     WEIGHT 20.

PTERM PT02DDS2
     TYPE VTAM
     BLOCKSIZE 8192
     TARGET DDSNODEX (NOT SHOWN ABOVE)
     WEIGHT 30.

PTERM PT02BULK
     TYPE BULK.

LTERM LT02BULK
     PTERM IS PT02BULK.
```

Note: VTAM minor node names must be unique throughout the VTAM network. Therefore, the APPL ID IS parameter on the VTAMLIN (L3270V) LINE statement and the SOURCE parameter on the DDS LINE statement cannot be the same even though these LINE statements participate in the same DC system.

Index

S

T

V

W